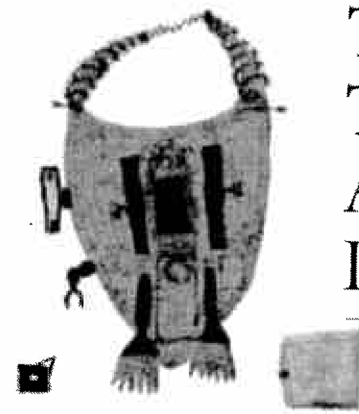
Teaching with Technology

A Model for Integration

Mary Lang, Ed.D.

Teaching with Technology: A Model for Integration



Problem Solving in a Mindful Way

Mary Lang, Ed.D.

Cover Graphic: Dylan Thompson

Sketches: Quinn Thompson and Ethan Kimberling

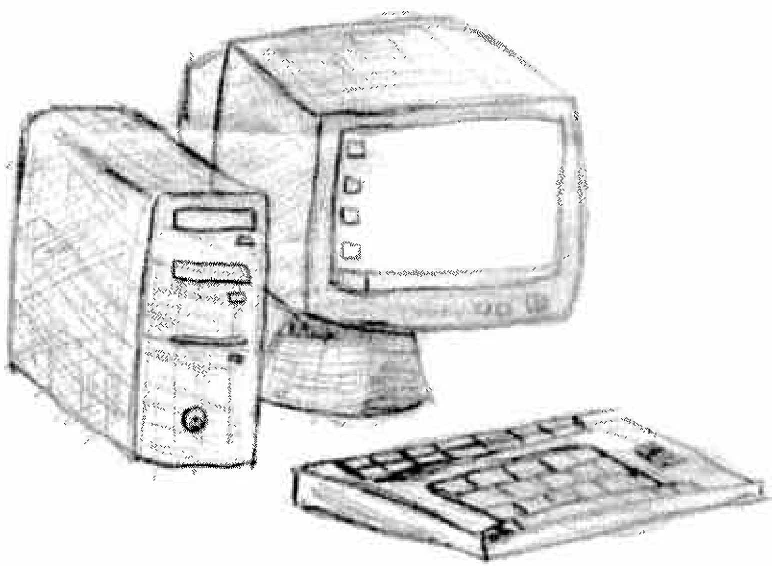
Editor: Julie Monroe

Designer: Melissa Rockwood, Rdesign

Thank you to the J.A. and Kathryn Albertson Foundation and the Idaho Department of Education for providing grants to Idaho public schools to fund the development of technology programs. Thank you to Melissa Rockwood for designing the cover of this book. Thank you to Julie Monroe for "going with the flow" during the editing process. Finally, my deepest appreciation goes to my husband Joe Thompson for sharing my vision.

To my children Dylan and Quinn, my reason for being passionate about education.

Table of Contents



Introduction

The Moscow Charter School in Moscow, Idaho is an accredited public elementary school serving grades kindergarten through sixth. Its mission is to provide an educational environment in which children's social, intellectual, and motor development is developed through a stimulating, well-rounded, hands-on, minds-on curriculum that integrates instruction in basic academic skills with creativity and problem-solving. Its unique approach to education offers parents in the community we serve a much-needed alternative to traditional educational practices.

In the spring of 1998, a founding group of parents and educators created a plan to serve as the foundation curriculum for the then-new Moscow Charter School. With this school, we wanted our students to master a variety of skills that would provide them with the tools to be good thinkers and to achieve successful intelligence. We knew that a large component of successful intelligence was based on a person's ability to find creative solutions to life's problems, thus in the beginning stages of our curriculum development, fostering creativity, problem-solving, and higher order thinking skills became a major goal of the school's curriculum.

Our combined philosophy was based on the notion that very young children are naturally strong in creativity and problem-solving as can be seen in their ability to progress rapidly from one developmental stage to another. Traditional educational practices do not necessarily encourage these types of behaviors into adulthood. Indeed, traditional educational curriculums can discourage the further development of these behaviors and thinking skills by completely focusing on activities that require memorization or continuous rote drill of information presented outside a meaningful context. Thus, in a traditional education environment, what was once innate behavior in all children that can be expanded upon with formal education, is left to atrophy through the predominant use of instructional practices that require rote [illegible] learning of information that is soon forgotten. Because [illegible] experienced this type of educational atmosphere [illegible] we understand the negative consequences of these poor educational practices. Our own experiences as students provided us with the inspiration to create an alternative education program.

The emphasis of the Moscow Charter School curriculum is also based on research, which demonstrates that children who are educated in diverse environments, experience the development of a wide range of neural synapses during critical periods of brain development. These critical periods are ages during which learning is optimal and efficient and receptive to specific tasks. For example, research demonstrates that children learn foreign languages easily and naturally below the age of seven. Each year over the age of seven, foreign languages become progressively more difficult to learn. It has been repeatedly demonstrated by researchers that laying a firm foundation for brain development in the early years will benefit the individual during an entire lifetime of learning and can even enhance recovery if the brain is damaged. (Diamond and Hopson, 1999; Jenson, 1998)

At the Moscow Charter School, we believe technology is an important tool in the general intellectual development of children that, if used properly, can encourage and teach successful intelligence skills in students. We view technology as a thinking/learning tool, a vehicle for creative expression, and a tool that will help students solve a variety of problems. Teaching students to use technology as a tool for communication, thinking, or problem-solving depends less on the type of equipment that a school has and more on how it is used. With this view, the student is seen as a programmer and uses technology much in the same way he/she uses a pencil. The teacher is also a programmer when he/she uses technology to support project-based learning. Our technology program also emphasizes both the preservation of individualized, self-paced learning within a group setting, as well as opportunities for group collaboration. We believe these goals represent the future in technology education.

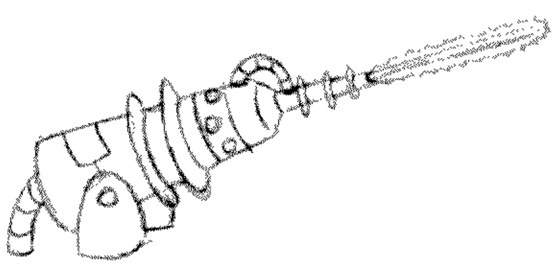
Purpose of this Book

This book is the second in a series of three books for educators and parents describing the Moscow Charter School's alternative approach to the education of young children. The series was funded through a charter school start-up grant from federal funds administered by the Idaho Department of Education. The first book, *The Arts in Education: A Model for Integration* shows educators how to prepare their students for success in the 21st century through the teaching of a curriculum in which the arts are integrated into instruction of the traditional academic disciplines. This book, *Teaching with Technology: A Model for Integration*, is the second in the series and describes the different components of the Moscow Charter School technology program, and the third book will examine the issue of financing charter schools.

Teaching with Technology: A Model for Integration serves an important purpose in today's highly computerized and connected world. Not only does it describes the technology program of the Moscow Charter School, it also explains, through detailed procedures, how other schools can adopt (or adapt) our approach and methodologies to the needs of their specific school. For that reason, this book is designed for educators and parents who wish to implement integrated technology experiences into their school's curriculum. With this book, I hope to provide them with a "blueprint" to facilitate that implementation.

Also in this book, I present the philosophical foundation for the Moscow Charter School's approach to technology instruction and explain that the rationale for our various curriculum directions is based upon contemporary educational research. I show how technology and technology instruction are used to achieve the mission of our school and how they support the unique features of our curriculum and further our goal of offering individualized education within a group setting.

Upon our philosophical foundation rests [illegible] tech-nology instruction at the Moscow Charter School [illegible]ed, I present the MCS technology mission statement [illegible] state-ments for achieving that mission, and the intended outcomes of our technology program. The core of this book [illegible] is the section that provides explicit instructions for the integration of technology into the curriculum. The individuals who founded the Moscow Charter School understood that while technology train-ing is essential for today's students, instruction is most effective when it is integrated into the instruction in the disciplines wher-ever appropriate. This is why so much of this book is devoted to the detailed presentation of specific activities, lessons, resources, as well as guidelines for the selection of software and hardware. In conclusion, I examine the implications of technology educa-tion for the future.



Philosophical Foundation for Teaching with Technology

After five years of using our integrated curriculum, we have observed that the combined emphasis of teaching basic skills, tech-nology, and the arts enhances creativity and promotes the learn-ing of higher order thinking and problem-solving skills for elementary age students. We believe this is because our program distinguishes between the use of technology as a tool with which to think and a curriculum environment in which technology pro-grams the child. This curriculum decision is based on Bloom's tax-onomy of higher order thinking skills, identified in Appendix A.

At the Moscow Charter School, we believe that what teachers need to know most about technology must be based on their knowledge of learning theory; therefore, the principles support-ing our technology education program are all based on learning theory. Seymour Papert, the author of LOGO, a programming language for children, as well as several books about children and technology, has heavily influenced our beliefs, which are:

• technology use should be based on knowledge of learning and brain development

• technology is an ideal tool for expression, thinking and prob-lem-solving

• intensive use of technology will change a child's thinking style and brain development; therefore, technology use in the class-room should be well thought out by the teacher

• technology should be used to teach children to think in better ways than they were in previous generations

• technological opportunities should be turned into learning advantages

• technology should not be used to "sedate" young children

• technology can be used to enable a child to find his/her personal path to learning or own natural learning style

• the role of the teacher teaching technology [illegible] conditions for the invention of knowledge

One of the long-term goals at the Moscow Charter School is to encourage each student to develop successful intelligence as well as to teach him/her to think more successfully than previous generations. We view higher order thinking and problem solving as skills that require left brain use of logic and organization along with right brain use of creativity and imagination. According to Sternberg (1997), having the ability to use these skills appropriately is a function of successful intelligence. "Successful intelligence is not just a cognitive ability – it's in large part a reflective attitude toward life and how one is living it." He suggests that if successfully intelligent people are not getting the results that they want, they are capable of using their problem-solving abilities to reassess aspects of the problem. Furthermore, Paul and Elder (2001) believe that the single most important variable in determining the quality of what a student learns is thinking ability. They believe that even the best teachers help students very little if their students lack the intellectual skills necessary for thinking well.

As a result of our beliefs, it is our goal to teach and use technology as a teaching tool during critical periods of brain development (K-6th). We teach a wide range of technology basics at the elementary level, just as we teach the basics in reading and writing. We also integrate technology as a problem-solving tool into our mathematics curriculum. We use technology purposefully; it is never used to "sedate" students or with students who learn more effectively with a different medium.

Technology as a Tool for Realizing the Mission of the Moscow Charter School

While learning theory provides the foundation for our curriculum decisions, which includes our technology instruction program, the Moscow Charter School also conducted a needs assessment during the beginning stages of its charter development. Our purpose was to ensure that technology instruction was integrated into the school's overall mission. The needs assessment also helped us determine the methodologies for using technology to achieve our curriculum goals and, interestingly, showed there was a need for a school in Moscow with our technology emphasis. Thus, the technology education program of the Moscow Charter School is unique.

Our program serves approximately 125 students and is based on a hardware platform of personal computers located in each classroom that are connected to a central server. The ratio of desktop computer to student is approximately three students to one computer for kindergarten through third grade, and two students to one computer for grades four through six. All students are given a user name that connects them to a roaming profile, which contains their saved files. We also have a laboratory of seventeen laptop computers contained in a mobile charging cart. These laptops are mobile because they are connected to the main server through a wireless connection. Therefore, they can be used in any classroom for any computer-based project.

With this hardware configuration, we use a project-based model that emphasizes problem-solving and critical thinking. With this approach, students and faculty come to understand the possibilities that technology can offer to enhance thinking skills and to take projects beyond the limits of the traditional classroom. Most significantly, our program presents information to students in an integrated format that facilitates discovery and emotional involvement during the learning process, which, in turn, [illegible] meaning and retention for the learner. (Rose, 1987) Other unique features of the Moscow Charter School curriculum also include the use of integrated technology instruction to enhance the student's learning experience in a meaningful way. These features include a self-paced learning environment, integrated thematic instruction, foreign language instruction, an emphasis on mentoring, the use of a school wide money system, and a mathematics curriculum with a strong problem-solving component

One of the things that make us unique is our belief in accommodating the learning level of every student. By using block scheduling and academic assessment, we have designed a curriculum that allows each student to experience self-paced learning in a structured group setting. Since all basic skills classes for each grade level are taught at the same time, students are placed in the appropriate class that will challenge them at their academic level for reading, writing and math. Self-pacing is inherent when technology is used properly in the academic environment. Choosing technology-based, self-paced activities supports the school's broader philosophy to provide students with individualized programming within a group setting.

Even though most of our students are enrolled in a specific grade level, it is important at the Moscow Charter School that students are offered a self-paced learning environment. For basic skills, individual students attend the class that matches their skill level; they then return to their homeroom for science and social studies as well as all special courses in the arts, foreign languages, environmental science, and physical education. Technology-based software tools, such as *Accelerated Math* and *Accelerated Reader*, are available to all grades at the school; they enable students to supplement basic instruction and advance in reading, math, science and social studies within a self-paced environment.

We also believe, however, that to produce a well-rounded student with versatile thinking skills other more creative educational opportunities must be offered that integrate basic skills with other meaningful activities. Learning basic skills in a meaningful way

provides the individual with a more versatile set of tools with which to solve problems and enhances the retention of information. Thus, at the Moscow Charter School, art, music, theater, dance and foreign languages are integrated into the standard curriculum in an effort to provide students with meaningful environments where they learn mindfully.

To achieve our goal of integrated thematic instruction, the entire school follows a year-long theme that serves as the a framework for basic skills integration. The theme, which is carefully chosen by teachers and staff, is nonetheless eclectic, which helps us study a variety of sub- topics. For example, the theme for 2003 was leaders and cultures around the world. Previous themes have included the Mars Millennium Project, the ancient Greeks, and Leonardo da Vinci. Each year students intensively research the chosen topic during the first three months of the year. The final six months are spent writing and preparing a theater presentation that demonstrates what we have learned about the topic.

Technology is incorporated into this approach. Students learn World Wide Web searching techniques in order to conduct research about the topic. Using word processing applications, students write essays and poems to provide the foundation for an original theater production that is written by teachers and students. Technology-based multimedia productions, including student created movie clips, are sometimes created and incorporated into the year-end theater production.

Another unique feature of our curriculum design is the emphasis we place on foreign language instruction. All students at the Moscow Charter School are taught Spanish. Our foreign language instruction, which begins as early as kindergarten, is based on neuro-physiological research, which demonstrates that children learn languages quickly in their early years. Students who are exposed to foreign languages during critical years of brain development, ages 1 through 12, will have an easier time learning languages later in life.

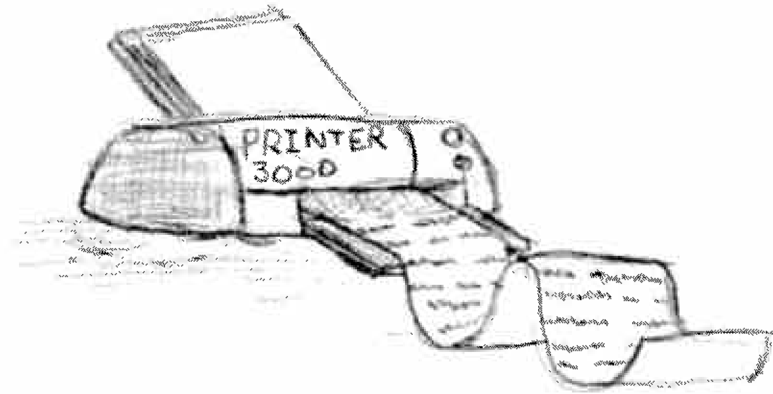
Besides Spanish, all Moscow Charter School students also learn a computer programming language starting in the third grade.

Students in the higher grades also learn basic programming skills to enable them to develop a website. Because programming languages are similar in nature to natural languages, learning a programming language further reinforces a student's ability to learn other languages. We have also observed that programming provides a rich environment in which students learn to approach a variety of problems with a variety of solutions.

Furthermore, at the Moscow Charter School, we have observed that technology facilitates student-to-student teaching and mentoring because it provides an environment that lends itself to these activities. Thus, at our school, students who grasp technology concepts faster than other students are encouraged to become "teachers" and teacher's helpers for other students. In addition, we also encourage students to complete projects as a group, with individuals assisting each other in achieving the intended outcome. This is significant because we have also observed that the greatest retention occurs when students mentor each other.

Besides foreign language instruction, one of the most notable aspects of instruction at the Moscow Charter School is the significance we place on creative problem-solving. Many psychologists define problems as falling in into two categories: well-structured and ill-structured. Sternberg's research on successful intelligence, suggests that in the real world successful intelligence is the ability to solve ill-structured problems, those without clear solutions. He says, "…an unfortunate failure of much education today, as well as the assessment of educational progress, is its overwhelming emphasis on well-structured problems."

However, many basic skills curriculums are designed to give students only problems with analytical solutions that are clearly defined. These types of curriculums teach students in a meaningful way only if the teacher follows up with a series of tasks that eventually enable the student to identify and to find his/her own solutions to related problems involving the subject at hand. In fact, teachers who randomly distribute unrelated worksheets to students or who focus entirely on memorization of knowledge may create a learning environment that encourages students to develop



a mindset that there is only one "correct" answer and that mistakes are to be avoided. In this type of environment, students learn to avoid solving problems because they interpret the experience of getting the wrong answer as a negative experience. Avoidance of or feeling negative toward problem-solving will prevent individuals from being able to define a problem clearly when they encounter it, thus further inhibiting their ability to solve it. Overuse of these types of curriculum materials in today's schools may be one of the reasons American students typically score low in the area of problem-solving on national standardized tests for mathematics. Encouraging students to develop a positive attitude toward problem-solving may be one of the most important behaviors that can be taught in formal school settings.

At the Moscow Charter School, we teach our students to realize that most real life problems are not solved neatly and that there are a variety of solutions to almost every problem. This is why we teach our students to think like programmers. An excellent example of the application of this concept occurs when students write an original computer program. The student almost never gets the intended outcome right the first time. An overall curriculum that encourages the notion that "if at first you don't succeed, try again" fosters a thinking style that approaches problems without fear and with an attitude that is open to finding multiple solutions. Learning to use creativity in one's approach to finding solutions will ultimately contribute to the future success of the individual.

Another example of the natural integration of technology training into students' overall learning experience is our school wide money system. This system teaches students three basic concepts: wage-earning, money management, and the use of technology to manage money efficiently. The participants in the program are students in grades three through six.

Each year, the Moscow Charter School prints its own currency in denominations of one, five, ten, twenty, and one hundred. Teachers regulate the distribution of money according to their own classroom plan. Generally, students earn wages by performing jobs within their classroom. The system also provides an opportunity for students to spend the money they earn. Each month, a different classroom prepares a market at which items created by the students in that classroom are sold to the entire school. The theme of the market is usually related to the year-long theme that is then being studied throughout the school. For example, this year the overall theme was cultures around the world, therefore, each class used a separate culture, such as Europe, Spain, or Antarctica, as the theme of its market. In preparation for the market, students studied the goods of that particular culture and created handmade products to sell on market day. All students were free to use their school money to buy products sold at the market.

One of the first things students do in the system is establish a bank within their classroom where cash and statements are stored. Each month, a different student is appointed to be the banker. The other students open accounts in the bank and maintain both handwritten ledgers and electronic spreadsheets to track their financial transactions. Students keep a running balance of their accounts and learn to use a checkbook, to read bank statements, and to use electronic spreadsheets to verify handwritten ledgers. Registers maintained as spreadsheets are updated at the beginning of each month, immediately after payday and the day the market is held. Students print out their spreadsheet bank statement and reconcile their checkbooks against the statement. We have found that through this program, students make a variety of important discoveries about money management and entrepreneurship.

The format for checkbooks and spreadsheets is shown below, along with sample data that might have been entered by a student:

BANK NAME

Name:
Address:
City/State/Zip:

Date	Transaction	Payment Debit (-)	Deposit Credit (+)	Balance
9/1/03	Payday		$30.00	$30.00
9/15/03	World Market	$20.00		$10.00
9/17/03	Fine	$1.00		$9.00
9/25/03	Reward		$5.00	$14.00

Envisioning the Integration of Technology into the Curriculum

Viewing technology as a tool for realizing the mission and curriculum goals of the Moscow Charter School provides the conceptual framework for our technology program. Yet, to achieve our overall mission and curriculum goals, it was necessary to develop a mission specifically for a technology education program.

The mission of the Moscow Charter School technology program is to use technology to enhance the intellectual development of each student, to prepare students to live and to work in an increasingly technological society, and to use technology as a tool to improve education delivery. In other words, the overall goal of our technology program is to use all forms of technology as tools to enhance and to extend education beyond the traditional classroom and library.

It is our belief that children develop attitudes about technology use as early as preschool. Therefore, we believe elementary schools are responsible for shaping the primary attitudes of both children and parents in the effective use of computers and other electronic technology. Further, we intend to teach children that computers are not devices that only entertain.

Research in brain-based learning supports our approach to technology (Jenson, 1994). Our emphasis on performing project-based tasks enables students to learn within context and to drive the pace of their own learning through the use of simulations, visualizations, and interactive software. In addition, the software and hardware utilized in our program encourages critical thinking and problem-solving skills, (e.g., programming and robotics), and develops communication skills, (e.g. word processing, multi-media, and telecommunications).

To achieve the goals of our technology program, we established

seven vision statements, and to achieve each vision statement, we identified relative instructional goals:

Vision Statement: To have relevant technology readily available for each child in the classroom.

Instructional Goal 1: Instructional computers are available in every classroom at the following student to computer ratios:

K-2nd grades: 3 students to 1 computer

3rd-4th grades: 2 students to 1 computer

5th-6th grades: 1 student per computer

Instructional Goal 2: Instructional, application, and tutorial software, as well as the World Wide Web and robotics, are available to support core curriculum courses in every grade. Problem-solving (robotics) and programming software (LOGO) will be available in grades 1 through 6.

Vision Statement: Technology is available for students to take the expression of their ideas beyond that which would have been possible in traditional classrooms.

Instructional Goal 1: The following technology components are offered to third through sixth graders to assist them in researching and expressing academic ideas.

World Wide Web availability on all computers

Communication with word processing, *Microsoft PowerPoint*, and *Microsoft Publisher*

Robotics

Computer programming

Problem-solving and design software

Multimedia tools (i.e., digital camera and scanner and software)

Instructional Goal 2: A project-based model is implemented for all students by providing technology hardware and software in the areas of visualization, research, robotics, programming, communications, and virtual reality.

Vision Statement: Teachers will be trained [illegible] between the expression of student ideas and [illegible] to implement these ideas into project-based [illegible]

Instructional Goal 1: All classroom teachers at the school have either a desktop or laptop computer available to support their instructional program. Teachers' computers have World Wide Web access and e-mail capabilities.

Instructional Goal 2: It is the policy of the Moscow Charter School to hire teachers who have a solid knowledge of technology use in education.

Instructional Goal 3: Workshops and summer programs are offered to Moscow Charter School teachers to assist them in recognizing connections and possibilities in a project-based model that uses state-of-the-art technology. Intel's *Teach to the Future* training will be offered to all teachers at the school. This workshop stresses the design of academic units based on Essential and Unit Questions. Unit questions frame a specific set of lessons and are designed to point to and uncover essential questions through the lens of particular topics and subjects. In the classroom, teachers pre-plan each unit by thinking about how that topic would benefit from technology integration. While teachers start the thought process by considering the established content standards for the unit, students are encouraged to communicate ideas creatively with the use of word processing, slide presentation software, desktop publishing software, and web site design.

Vision Statement: Students will be able to drive their own rote learning tasks with the opportunity to move ahead in their learning at their own pace.

Instructional Goal 1: The following self-paced reading and math software is currently used at appropriate age levels.

Waterford Early Reader Program

Breakthrough to Literacy Program

Accelerated Math

Accelerated Reader

Commercial software programs in all subject areas are available as well.

Vision Statement: Students will devote time to in-depth thinking, collaborative study, and creative problem-solving using computers and robots.

Instructional Goal 1: Robotics programming is offered to all students in grades 1-6. This component consists of interactive software that teaches students to make logical and sequential decisions that cause machines to operate. Following the completion of this introductory component, students have opportunities to program three different robots.

Instructional Goal 2: A problem-solving and design course is offered to all students from grades 3 through 6. The course consists of solving problems by developing a model of the solution with concrete materials. Students are taught to simulate the problem and solution by using software that is appropriate for developing a solution.

Vision Statement: Technology at the Moscow Charter School will increase students' options and opportunities for individualized study. It will help motivate students and inspire ownership of the learning process. The technology program will empower students to move beyond traditional textbook learning to a world of information and discovery.

Instructional Goal 1: All students at the school have access to the World Wide Web and will receive classroom training on how to use it.

Instructional Goal 2: Students are encouraged to author their own presentations and lessons with the LOGO programming language, *Microsoft PowerPoint*, *Authorware*, and *Microsoft Publisher*.

Instructional Goal 3: A website is currently being developed to allow gifted and talented students to identify on-line courses on topics of their own choosing in which they may wish to enroll.

Vision Statement: Computers will serve students as tools to continue their self-directed studies into the summer months.

Instructional Goal 1: Technology courses will be offered to students from the Moscow Charter School and surrounding schools by the Moscow Charter School faculty during the summer.

To achieve our mission, vision statements, and instructional goals, we have established the following objectives.

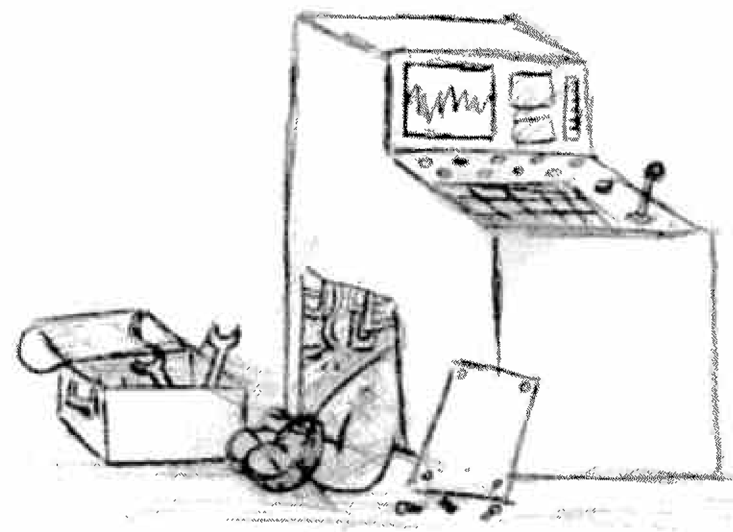
1. To maintain an infrastructure that supports the design of our technology network structure, we continually purchase state-of-the-art technology that is retrofitted to keep up with rapid changes in hardware, conduct yearly evaluations and upgrades of hardware and software, prioritize technology budget funding to allow for infrastructure costs on a yearly basis, work with network designers to plan future costs, and manage security issues through infrastructure configuration.

2. Our technology committee develops policy and procedures to keep computer systems operating efficiently. We regularly inform staff and students of policies and develop and increase training and communication with parents, the local universities, and with the international community.

Through the realization of these goals and objectives, the intended outcomes of our technology program are:

- Technology will be readily accessible in the classroom to all students at the Moscow Charter School. Third through sixth grade students should be able to support their learning with technology anytime throughout the academic day.
- Moscow Charter School students will focus on critical thinking, problem-solving, communication, and research with software applications, robotics, and Internet access readily available at every grade level.
- Students and faculty are trained to understand the possibilities that technology allows for enhancing thinking skills and

creativity and taking projects beyond the limits of the traditional classroom.
- The technology committee at the school will continue to provide input to teachers and present training programs for the recommendations described above.



Selecting and Using Software: An Overview

At the Moscow Charter School, our technology program is based on a philosophical foundation of brain-based research that guides our overall curriculum decisions. Upon that foundation we built a framework of vision statements and instructional goals and objectives to guide our methodologies of instruction. In turn, this framework guides us in the selection and purchasing of educational software and the hardware through which that software is used.

This section of Teaching with Technology serves as a primer for educators faced with the challenge of purchasing computer software. We use a variety of software applications for all different subjects at the Moscow Charter School. One of the most important considerations in choosing software comes from our awareness that much commercial software favors quick reactions over long-term thinking in an effort to capture and to maintain a student's attention. For this reason, we emphasize the use of programming languages for student use. We feel this emphasis encourages students to use technology to develop higher order thinking skills, not for purposes of "drill and skill" or entertainment. We also avoid software that has a video game-based format, especially in the early primary grades, because research shows that the fast-paced visual formats of video games develop neural skills contrary to those that are necessary for learning to read.

Teachers at the Moscow Charter School are trained to evaluate software, and programs are carefully chosen with knowledge of their specific purposes. During our evaluation of software applications, we pay particular attention to the answer processing component portion of the program. Software that gives the student meaningful feedback about a specific response is chosen over less responsive software. We encourage the use of software that teaches students the process for solving a problem they have answered in error, not software that tells the user the correct answer. The ultimate criterion, however, is put in place when teachers ask themselves if a particular software application is the best possible way to teach the topic at hand.

Below is a matrix showing the software the Moscow Charter School has chosen to achieve our curriculum goal of improving reading, writing, mathematical, thinking and logic skills through the integration of appropriate pedagogical software. The matrix is organized by grade level and includes very specific information about the software we selected, such as its brand name. I also explain the software application's broad purpose. Following the matrix, I describe in more detail the types of software as organized by the specific outcomes they produce.

• ***Reading and Language Arts***

Class	Software
Kindergarten	*Waterford Early Reading Program* *Breakthrough to Literacy* Accelerated Reader Basic skills software
Grades 1 and 2	*Breakthrough to Literacy* *Accelerated Reader* Word processing Reading/writing instructional software Keyboarding
Grades 3 through 6	*Accelerated Reader* *Microsoft Word* *Microsoft PowerPoint* *Microsoft Publisher* World Wide Web access and research *Authorware* *Hyperstudio*

• *Mathematics*

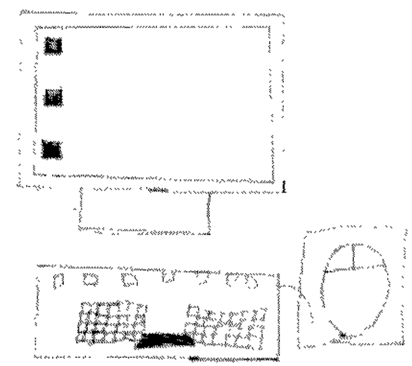
Class	Software
Kindergarten	*Roamer Robot* Math instructional software
Grades 1 and 2	*Accelerated Math* *Roamer Robot* *Crystal Rainforest*
Grades 3 through 6	*Accelerated Math* *Roamer Robot* *Crystal Rainforest* *Mission Control* *Excel* Spreadsheet *LOGO* programming language *LOGO* robot arm *LEGO* mindstorms robot Mathematics interactive software Database management software

• *Science*

Class	Software
Grades 1-6	World Wide Web research *Microsoft Word* *Microsoft PowerPoint* *Microsoft Publisher* *Excel* Database management software Virtual physics software Science interactive software

• *Social Studies*

Class	Software
Grades 1-6	World Wide Web research *Microsoft Word* *Microsoft PowerPoint* *Microsoft Publisher* *Excel* Database management software Social Studies interactive software



Descriptions of Software

Drill and practice software practices and reinforces the learning of specific facts, such as multiplication or sight word recognition. This type of software most often favors quick reactions over thinking skills and often uses a video game format. The advantage of this type of software is that students can pace themselves in their learning of specific facts and experience necessary repetition of similar problems.

Drill and practice type software packages without the above-mentioned weaknesses do exist. Examples include *Accelerated Math* and *Accelerated Reader* published by Renaissance Learning. All students at the Moscow Charter School use these self-paced reading and mathematics software applications to supplement and to reinforce their knowledge of math and reading.

With *Accelerated Math*, students take a test to determine his/her level of math proficiency for a wide variety of math concepts. Math worksheets covering skill-related mathematical concepts appropriate to the student's performance on the test are printed out. After completing each worksheet, the student scans his or her answer sheet. The computer software then evaluates the answers and prints out a new mathematical worksheet based on the student's score. The software tests students periodically to determine their exact level of proficiency. This type of software is used as a small component of our overall math curriculum.

Accelerated Reader promotes reading of books that are within a student's reading level. When using *Accelerated Reader*, the student is tested for comprehension on a computerized test. The student begins by taking a test of reading skills and comprehension using the STARR reading program. *Accelerated Reader* then produces a printout of suggested books within the user's reading range. After the student reads a book listed on the printout, he or she takes a comprehension test on the computer. Based upon the

test results, another list of suggested readings is produced. Both *Accelerated Reader* and *Accelerated Math* software programs provide progress reports for teachers.

Other types of drill and practice software used at the Moscow Charter School include software to teach keyboarding skills. These skills are first introduced at the first grade level.

In addition to drill and skill software, the Moscow Charter School also uses **simulation software**. This type of software presents a meaningful scenario while allowing the user to input solutions to problems they may encounter along the way. We support the use of academically relevant simulation software at the Moscow Charter School because of its emphasis on ill-structured problems. Students rarely get experience in solving ill-structured problems and gain decision-making experience.

Examples of this software are the popular *SIMS* series that includes *SIM Family* and *SIM Theme Park* where users build their own theme park and learn to make a variety of business decisions, such as hiring and firing employees and purchasing equipment, that determine the success of their enterprise.

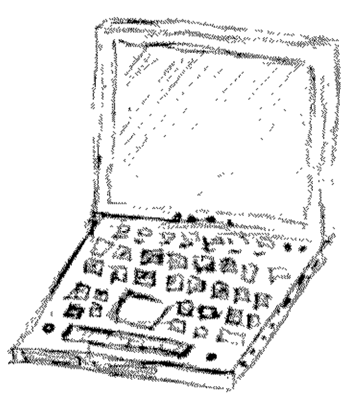
Two simulation applications we use are *Crystal Rainforest* and *Mission Control*. They are used to prepare students to enter the programming environment in our technology component of the math curriculum. *Crystal Rainforest* is a simulation game that allows students to make decisions within a story environment using LOGO commands. LOGO is a programming language written by Seymour Papert in the 1980s specifically for use by children. We teach LOGO to all students from third through sixth grade. *Mission Control* teaches the user to program simple robots by making a series of decision that produce a simple vending machine or gear mechanism. Successful decision-making is rewarded when the machine produced works.

Another type of software used at the Moscow Charter School is **tutorial software**. This type of software teaches students factual information in a tutoring format similar to a book except that pacing is based on achievement. We use two specific tutorial software programs at the kindergarten and first grades to teach reading and phonics.

The *Waterford Early Reading Program*, developed by the non-profit Waterford Institute, is used in kindergarten. The Waterford program is one component of an integrated reading program that includes big books, small books to take home, and corresponding videos to teach beginning reading skills and phonics. The Waterford software is designed to give each kindergarten student 15 minutes on the computer each day for phonics practice. Students work through a hierarchy of phonics lessons. Progression through the software is based on student achievement.

The *Breakthrough to Literacy* software published by McGraw-Hill is also a component of a larger program that includes big books, small books to take home, and corresponding activities designed to teach first-graders reading skills. The Breakthrough software also takes the student through a hierarchy of phonics lessons that is self-paced, with advancement based on student achievement.

Another type of software frequently used at the school is **application software**, the type of software most often used by adults in their work environments. Application software is used for a specific purpose, for example, word processing, presentation and graphics, spreadsheets, and database management. This type of software can easily be used to encourage creative thinking and the development of higher level thinking skills.



Students use word processing applications when developing writing skills, and keyboarding skills are started as early as the first grade. Students and teachers learn to use spreadsheet applications in the third grade. Teachers use spreadsheets to track projects, and our school-wide money system is an example of how students use spreadsheets to manage numerical information. Students also use spreadsheets as a communication tools. For example, spreadsheets are used to account for scientific data that is collected for science projects. Students design the format for the data, collect it, and then input their results into a spreadsheet. The spreadsheet software enables students to transform the raw data into graphs, which effectively communicates results and trends.

A related example of using application software to manage and communicate data is the participation of all fifth and sixth grade students in the McCall Science School held at the University of Idaho's research station in McCall, Idaho. Using a spreadsheet application, students collect data, record it in the spreadsheet, and then transport the data into *Microsoft PowerPoint* to make an integrated presentation of their experiences to other students and parents at the school.

At the Moscow Charter School, we also incorporate database management software into a lesson if students need to organize and retrieve data in a systematic way. Students are taught to keep a database that contains the names, addresses, and telephone numbers of their classmates.

At the Moscow Charter School, we encourage students to use communications software to make multimedia presentations as tools if they increase a student's understanding of content through the presentation. If this criterion is met, students use *Microsoft PowerPoint* and *Microsoft Publisher* to develop multimedia presentations on a variety of topics. For example, fifth and sixth grade students participating in the McCall Research Center science school use *Microsoft PowerPoint* to develop presentations about the ecosystems they've studied; the presentations include graphics, animation, and sound imported from the World Wide Web to enhance these presentations. Students are ass[illegible] their presentations by first developing a storyboar[illegible] orga-nizing the material. An outline of the storyb[illegible] devel-oped directly on *PowerPoint*.

In addition, teachers frequently use *Microsoft PowerPoint* in the classroom in a variety of ways, including the presentation of charts and tables, survey and questionnaire results and science fair projects. This software can also be used to incorporate research from a CD-ROM reference library or the World Wide Web into a lesson or to develop a personal picture storybook.

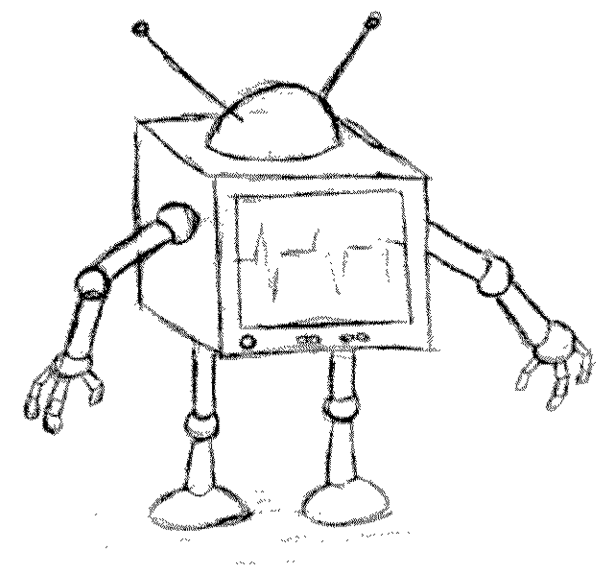
Another Microsoft application, *Microsoft Publisher* is used to produce a variety of communication products, including news-letters, flyers to inform parents of school happenings, a teacher's syllabus designed as a calendar, weekly calendars for students to keep track of classroom events and projects, business cards, pro-motional materials, invitations, posters, menus, websites, and newspapers. Like the multimedia presentations, students use storyboards and outlines to plan their projects.

Students also use *Microsoft Publisher* and *Microsoft PowerPoint* to develop websites that meet the targeted learning goals of a class or a specific project. Each year, fifth and sixth grade students in Idaho are invited to participate in the Mars Rover state competi-tion sponsored by NASA to design and build a robot with a vari-ety of characteristics. In this project, students develop a web page that outlines their progress and participation in this competition. At the Moscow Charter School, we use *Microsoft Publisher* to cre-ate simple websites quickly through the use of its Website Wizard.

To enhance their presentations, students at the Moscow Char-ter School have use of digital and video cameras. Students down-load files taken with a variety of brand name cameras onto a hard drive or server. These files can then be cut and pasted into a num-ber of different applications, such as word processing documents, brochures, and web pages. Students at the Moscow Charter School use a LOGO video camera to film video animation clips. The soft-ware attached to this camera allows for frame-by-frame editing,

and these video animations have been very popularly used in our end-of-the-year theater production; they provide us with a true multimedia production.

In addition to software stored locally, teachers and students also use Internet browser programs to locate resources for specific purposes. While the Internet is a powerful tool for teachers, they are mindful of the importance of teaching students who are be-ing taught to use search engines and reference sites, such as *Microsoft Encarta*, to follow "fair use" guidelines and to obtain materials from the World Wide Web lawfully. Students are also taught to cite all references and to evaluate Web resources. See Appendix B for a complete outline of our curriculum for teach-ing Internet research and tips for obtaining graphic and sound clips from the World Wide Web.



Using Programming Software and Programming with LOGO: Specifics

The Moscow Charter School is committed to giving its students the tools they need to master the complexities of life in the Information Age. Creativity, flexibility, problem-solving, and resiliency are all skills that characterize successful intelligence. We believe programming software and learning to write a computer program using the LOGO language are excellent tools for developing these sorts of skills; therefore, in this section, I provide a detailed description of the Moscow Charter School's use of this type of software.

While at the Massachusetts Institute for Technology in the 1970s, Seymour Papert wrote the LOGO programming language. Papert had completed an extensive period of study with Jean Piaget prior to writing the language, which significantly influenced his decision to write a language specifically for children. LOGO is a list processing language, meaning that commands are stored in lists within the program. The commands found within the LOGO language are similar to or the same as English words; therefore, it is easy for children to learn. LOGO is also called turtle geometry, a different style of geometry that is computational rather than logical or algebraic.

Programming languages contain similar features of most natural languages in that they have no threshold or ceiling. Learning a programming or foreign language in their early years enables students to use the portion of their brain that learns a new language during critical periods of brain development. Thus, we believe programming is the ultimate tool for teaching both creativity and problem-solving because it encourages students to create their own solutions from an original idea. Learning to write a computer program is more about problem-solving than it is about technology, and problem-solving is inherent in the act of programming.

To better understand the role that programming plays in brain development, it is important to understand the process of solving problems. Sternberg (1998) identifies six basic steps that constitute a cycle of problem-solving. These include:

1. Problem recognition
2. Problem definition
3. Formulating a strategy for problem-solving
4. Representing information
5. Allocating resources
6. Monitoring and evaluation

By teaching the LOGO programming language, students gain experience with every aspect of this problem-solving cycle. When we teach LOGO, we help students identify aspects of problems while giving them practice in defining the problem once it is identified. We realize this gives students who may be highly skilled at solving analytical problems experience in solving ill-defined problems. During the programming process, we also attempt to prevent mental sets and fixations (which hinder problem-solving) by working specifically on identifying problems and interesting solutions. With LOGO, even though teachers are teaching a systematic procedure for thinking about thinking, they are also creating a learning environment that provides an open-ended approach that encourages the development of multiple solutions

Papert observed that children who were required to program computers used very concrete models to think about thinking and to learn about learning. In doing so, they become experts on thinking about their own learning strategies. By defining the role of the student as a technology programmer, we teach students valuable learning strategies and higher-order thinking skills.

At the Moscow Charter School, we begin preparing students to learn programming languages early on. In the first grade, they are taught to program simple robots using software that encourages them to solve problems that are the same or similar to programming problems. By third grade, students are ready to learn LOGO. By the time they reach the fifth and sixth grades, they are ready to design, create, "debug," and test their [illegible] with their fellow students serving as users.

We use robots and LOGO because they [illegible] with an object with which to think. For example students relate the turtle robot or the turtle object on the computer screen to their own bodies, thus abstract thinking becomes more concrete as the student progresses in his or her ability to solve problems. Because LOGO is based on an environment that combines aspects of algebra and geometry, students also learn mathematical facts and concepts in the process of learning to program.

At the Moscow Charter School, students follow three steps to achieve expertise in computer programming. First, they are introduced to programming through a robotics program called *Roamer Robot*. With this program, students learn to program a moving object by pressing icons that correspond to particular commands. Next, students master the *Crystal Rainforest* and *Mission Control* software programs. These software packages enable students to think like programmers by solving problems to achieve a particular goal. *Crystal Rainforest* uses LOGO commands. Lastly, students use LOGO to write a program. In this stage, students design programs that teach basic concepts in geometry, algebra, and physics. They begin by developing a plan, and then program the appropriate commands to carry out the plan. They are also responsible for debugging their problems in order to produce a program that can be successfully used by others.

Step 1: Programming with the *Roamer Robot*

At the Moscow Charter School, the process of learning to program begins with teaching students to program a robot through the use of *Roamer Robot* software. The robot is a tangible object (portrayed as a turtle, for example) with a head and a tail. The tangibility of the object is important because it simulates the physical movement of a person, which in turn, helps students solve simple problems by conceptualizing how they would solve the problem with their own bodies. Students program the robot by pressing a sequence of icons on the turtle robot's back. By doing

so, they use the same commands and processes that are used in LOGO; thus, the concepts and commands students learn when using *Roamer Robot* easily transfer to the computer-programming environment. The software's lessons are structured not only to introduce students to basic LOGO commands but also to problem identification and strategic thinking. After students have mastered the basic commands and demonstrate they have a working knowledge of how to operate the robot according to a predetermined plan, they are ready to move through *Crystal Rainforest* and *Mission Control* and begin programming with LOGO.

To teach students to use *Roamer Robot*, we developed the following procedure:

1. Identify the Roamer's head and tail.
2. Use the basic commands to make Rover move.
3. Use basic commands to play music.
4. Encourage students to think of a programmable idea – a visual shape such as a square or triangle.
5. Use a series of commands to carry out the idea.
6. Encourage problem identification and debugging.
7. Programming a simple procedure.
8. Learn about nesting procedures (calling procedures from within other procedures).
9. Design small group work with Rover and Pen: 3-5 (students in the group). The pen is optional and fits into the body of the robot so that when the robot moves it can draw a shape as it executes a procedure.
   a. Send the rover to another person. For example, a student writes code that will send the rover to another student using only straight lines. Any variation of this idea works.
   b. Draw a projected path
   c. In advance, identify a set of commands that will realize the plan.
   d. Write down the commands, enter them into the robot, check them off as the robot executes each command, and "debug"

any errors. "Debugging" is complex, so we [illegible] a procedure that facilitates the identification of [illegible] makes the process much more manageable. We [illegible] have one person write down the code as another person enters the commands. As the robot successfully executes each command, the assistant checks them off.

10. Program a more complex procedure. For example, the robot goes into each classroom, plays a tune, pauses and then turns around and goes down the hall to the next classroom.

Step 2: Introducing programming skills

Students become acquainted with the LOGO programming environment by using the simulation software *Crystal Rainforest* and *Mission Control*. *Crystal Rainforest* is LOGO-based in that the user navigates a turtle-like object around the screen to achieve a particular goal. The object is moved using the same commands that exist in LOGO, such as FD (forward), BK (backward), RT (right), and LT (left). While using this software, users get experience with important concepts such as reading maps, determining direction and angles, and strategic thinking. *Mission Control* is a software game that allows the user to program simple robots. While using this software, students gain experience with strategic planning, logical thinking, and debugging – all skills that will benefit them in the programming environment.

Step 3: LOGO Programming

Writing LOGO code is the third step students take in our plan for teaching creativity and problem-solving through instruction in programming. LOGO fosters problem-solving at every level because the user rarely gets the program right the first time. Because the user can identify his/her own body with the object on the screen, even young children can use this concrete thinking style to solve problems. As problems become more and more sophisticated, students use this concrete reference point as a jumping off point for more sophisticated solutions. Papert says the use

of LOGO "allows us to shift the boundary separating the concrete and formal," thus challenging Piaget's claim that there is a distinction between a child's ability to use concrete and formal operations in their thinking. When children use the computer as an object with which to think, it helps make a formal thought concrete. In addition, while LOGO is a model for thinking, it also contains mathematical information and thus provides students with an environment that gives meaning to mathematics.

At the Moscow Charter School, we use two different LOGO versions to teach programming. These include version 2.0 and *Terrapin LOGO*. Both versions contain all of the commands listed in Appendix C, which details our LOGO teaching methods. However, *Terrapin LOGO* has an extensive graphics window that allows the user's LOGO program to become a true multimedia experience. Graphic features include bitmaps, animations, and interesting background scenes. Generally, students start with LOGO 2.0 to learn the specifics of the programming language, and when they are ready to program an original creation, *Terrapin LOGO* can then be used to add the "bells and whistles."

We use a conceptual approach to teaching programming. The basic concepts are listed in the outline below. Each component of the outline is presented in Appendix C with sample procedures.

Concepts Outline for Teaching LOGO

I. Identify an object with which to think, for example, your own body.

   A. Write LOGO commands in the immediate mode, which means that as soon as a student enters a command, it is executed by the LOGO turtle on the screen.

II. Understand shapes and symmetry:

   A. Define a polygon

   B. Write a basic procedure (a set of commands that can be saved as a program) and then "debug" it..

   C. Use the repeat command

D. Determine an angle with a formula [illegible] tri-angle that totals 360 degrees

III. Approach problem-solving in a systematic [illegible] good programming strategies, including [illegible]
A. Break down a problem into different components.
B. Use strategic thinking and organization
C. Nest procedures, meaning call a procedure from within another procedure.

IV. Use coordinates.

IV. Use a variable.

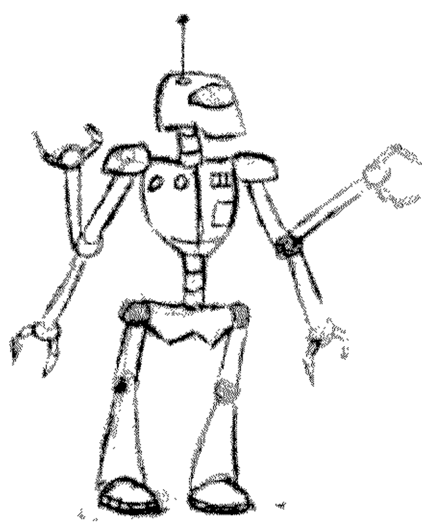
V. Use recursion or looping, also know as "IF THEN" logic.

VI. Program with text.

VII. Make arrays.

VIII. Design a program.

IX. Program an original program.



The Future of Technology Education at the Moscow Charter School and Beyond

At the Moscow Charter School, students use technology to achieve specific outcomes while developing cognitive skills that will help them become successful adults. This is achieved through the integration of technology use and instruction into the overall curriculum of the school, just as the arts are integrated into instruction in the basic academic skills. This holistic approach to education is fundamental to the Moscow Charter School; we follow it because we believe it best serves our students.

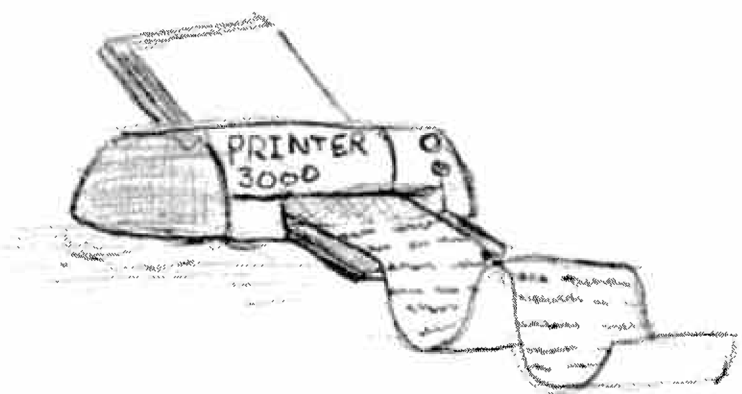
Because we believe so strongly in the concept of integration, we have applied for grant funding to formalize our art curriculum design into a program entitled **Integrated Arts Instruction and Student Achievement** (**IAISA**). If we receive funding, the IAISA program will enable us to use our curriculum design as a model for other schools interested in integrating the arts into their core curriculum. Technology, as a means of delivery, will be essential to the success of this program.

Integrated Arts Technology Laboratory

A major component of the IAISA program is its integrated arts technology laboratory, which will facilitate the effective implementation of the standards for arts and humanities education in a comprehensive and economical manner. As proposed, the hardware component of an integrated arts technology lab will consists of 16 computers and 16 MIDI keyboards that are interfaced via a media lab controller, connected to a digital recorder, and linked to the Moscow Charter School server. This lab will support the most current software available for instruction in a variety of subject areas, including music, art, foreign language, history, as well as science and language arts—the possibilities are unlimited. These disciplines may be taught as stand-alone subjects or with an interdisciplinary and integrated approach. The lab will facilitate individual or group instruction in a variety of formats, including direct teacher instruction, computerized instruction, or on-line instruction.

In music, for example, the integrated arts technology lab will promote musicianship and creativity. The technology lab will allow for class piano instruction enabling every student to receive piano lessons. In addition, there is a wide array of software programs for music education, specifically in the areas of early musician development, music-reading skills, listening and performing skills, music theory, keyboard studies, jazz and improvisation, music appreciation and history, music and culture, and music composition and notation. These software programs are interactive and effectively stimulate, motivate, and reinforce student engagement and achievement of learning potential and performance in music. Assessment of student progress and achievement is built into those software programs that allow for individualized monitoring and reporting. Digital recording and the production of student CDs will record and document the performances and compositions of each student. Thus, the integrated arts technology lab will provide a well-rounded music curriculum that is sequential and designed for the successful achievement of the national and state standards for music education.

The integrated arts technology lab will also be a conduit to the information highway. Such technology will allow teachers and



students to participate in the current trend of on-line curricula and web-based instruction. Educators around the world are utilizing the World Wide Web to pool instructional resources, share curriculums, and to establish an international and multicultural teaching and learning community. Web-Based Instruction (WBI) is the wave of the future for education, and integrated arts instruction is at the forefront of this movement. Through the development of this laboratory as part of the IAISA program, Moscow Charter School will not only benefit from interacting with the web-based educational community, it will also contribute to the community. Our proposed IAISA program includes a WBI component that will offer the full curriculum, instructional goals, and lesson plans of our integrated arts curriculum to any educator who may wish to adopt it.

Website for Gifted and Talented Students

In a separate project, we will develop a website to expand coursework possibilities for gifted and talented students. With this website, gifted and talented students will be able to access information about on-line courses in their specific area of interest. Instructions on how to sign up for on-line courses will be available as well. The site will also suggest specific simulation software in math, language arts, physics, chemistry, technology, and the arts.

Video Conferencing

Another initiative we have planned is the establishment of a curriculum delivery system via video conferencing. At the Moscow Charter School, we view video conferencing as a tool that bridges any knowledge gap that may currently exist because it enables us go beyond the limits of classroom walls.

Video conferencing is two-way video and audio that facilitates interaction between participants in two or more sites and allows for the sharing of documents or applications for purposes of instruction, communication, and collaboration. It is a very efficient delivery system that bridges distance gaps and saves schools time and money. In fact, some projects are enhanced through video conferencing. Our goal is to create a video conferencing system that will:

- Deliver courses for gifted and talented students
- Train teachers
- Serve as a bridge between classrooms or a classroom and a professional knowledge base
- Facilitate student-to-student mentoring and teaching from different locations
- Bring professionals into the classroom
- Provide training for special needs students when resources are not available locally

Conclusion

At the Moscow Charter School, our vision of technology education in the classroom is unique. The purpose of our technology education program is to support and to encourage young minds to discover new and creative uses for technology and to help them understand the trends that technology will take in the future. In addition, our technology education program also provides direct and indirect academic support through the use of interactive software, which accelerates and improves academic skills in all standard courses of study. Our technology education program also teaches problem-solving and higher-order thinking skills. Graduates of the Moscow Charter School enter the secondary grades with a firm foundation in the use and purposes of technology. As a result, they are able to express their creativity in completing class work and build upon their foundation of technology skills for use in the future.

Bibliography

Bancroft, W.J. "Suggestopedia, Sophrology and Traditional Foreign Language Class." *Foreign Language Annals*, 15, 1982, pp. 373-379.

Caine, R.N. and G. Caine. "Understanding a Brain Based Approach to Learning and Teaching." *Educational Leadership*, 28, 1990, pp. 66-70.

Caine, R.N. and G. Caine. *Making Connections: Teaching and the Human Brain*. Alexandria, VA: Association for Supervision and Curriculum Development, 1991.

Diamond, Marian and Janet Hopson. *Magic Trees of The Mind*. New York: Penguin Putnam, Inc, 1999.

Gardner, Howard. *Frames of Mind*. New York: Basic Books, 1983.

______. *Multiple Intelligences*. New York: Basic Books, 1993.

______. Gardner, Howard. *The Disciplined Mind*. New York: Simon and Schuster, 1999.

Goleman, Daniel. *Emotional Intelligence*. New York: Bantam Books, 1995.

Jensen, E.P. *The Learning Brain*. Del Mar, CA: Turning Point for Teachers, 1994.

Jenson, E.P. *Super-Teaching*. Del Mar, CA: Turning Point for Teachers, 1988.

Langer, Ellen. *The Power of Mindful Learning*. New York: Addison-Wesley Company, Inc., 1997.

Lozanov, G. *Suggestology and Outlines of Suggestopedy*. New York: Gordon & Breach, 1978.

Ostrander, Sheila and Lynn Schroeder. *Supermemory, the Revolution*. New York: Carroll & Graf, 1991.

Papert, Seymour. *Mindstorms: Children, Computers and Powerful Ideas*. Cambridge, MA: Perseus Publishing, 1993.

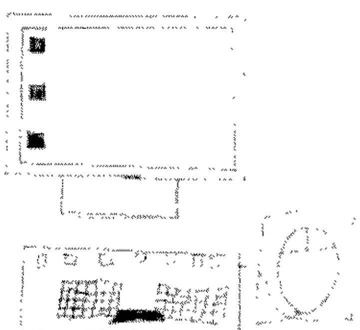
______. *The Connected Family*. Atlanta, GA: Lon[illegible] 1996.

Paul, Richard and Linda Elder. *Critical Thinking: [illegible] Taking Charge of Your Learning and Your Life*. Prenti[illegible]

Rose, Colin. *Accelerated Learning*. New York: Dell Publishing, 1987.

Schroeder, Lynn and Sheila Ostrander. *Superlearning*. New York, Dell, 1981.

Sternberg, Robert J. *Successful Intelligence*. New York: Penguin Putnam Incorporated, 1997.



Appendix A:

Bloom's Taxonomy for Higher Level Thinking Skills

Knowledge: Recall or recognition of information

arrange	define	duplicate	identify	label	list
memorize	name	recall	recognize	repeat	show
state					

I. Comprehension: Understand or Interpret

classify	compare	demonstrate	describe	differentiate
discuss	explain	express	identify	indicate
interpret	locate	paraphrase	report	summarize
restate	recognize	review	select	translate
visualize				

II. Application: Transfer from one setting to the another

apply	calculate	choose	classify	demonstrate
dramatize	illustrate	interpret	manipulate	modify
operate	relate	schedule	solve	use
write	put into practice			

III. Analysis: Identify parts and see related order

analyze	appraise	calculate	categorize	choose
compare	contrast	criticize	deduce	examine
experiment	organize	question	test	differentiate
discriminate	distinguish			

IV. Synthesis: Put parts together to from a new whole

arrange	assemble	compare	collect	compose
construct	create	design	devise	develop
discuss	formulate	hypothesize	manage	organize
plan	prepare	propose	report	set up
schematize	support	write		

Appendix B:
Locating Resources for Unit Portfolios

Teachers follow this outline in the Intel "Teach to Future" workshop manual to teach the following topics:

Using and understanding Directories

World Wide Web directories are used to seek information on a broad topic; an example is the Internet Public Library: <http://www.ipl.org>. To use a directory:

1. Select one of the directory's broad categories and narrow the focus step-by-step.
2. Search using a keyword (e.g. American wars) and continue clicking through to link to more specific topics
3. Results can vary from one directory to the next.

Using and understanding Web Search Engines

Search engines are used to access specific information when titles, phrases, or technical language are known. They are best used to locate a specific piece of information, such as a known document, rather than a general subject. A sample website that provides information on specific search engines is Search Engine Watch, <http://www.searchenginewatch.com>. Examples of search engines include:

Alta Vista: http://altavista.com

Google: http://google.com

Meta search engines search multiple databases simultaneously and give a report of its findings. Examples of meta search engines are:

Ask Jeeves: http://www.ask.com

Ask Jeeves for Kids: http:// www.ajkids.com/

Awesome Library: http://www.awesomelibrary.org/

Berits Best: http://www.beritsbest.com/

Kids Click: http:sunsite.berkeley.edu/kidsclick!

Kid's Search Tools: http://www.rcls.org/ksearch.htm

LycosZone: http://www.lycozone.com/

Super-Kids: http://www.superkids.com/

Yahooligans: http://www.yahooligans.com/

Specialty Search Engines

If searching for specialized information, a specialty search engines can be used. GovSpot, <http://www.govspot.com>, is an example of specialty search engines.

Narrowing Your Search

To narrow a search, use Boolean logic operators, AND, OR, and NOT. Searchers may also use the actual word or symbols that correspond to the words. Entering an equation, such as "+oatmeal+cookies-raisins" will produce a list of sites with information about oatmeal cookies without raisins. In addition, a phrase can be searched by enclosing it within quotation marks, e.g. "the only thing man has to fear is fear itself".

Citing a Website

The Modern Language Association suggests the following format when citing a website:

Last name of author, First name of author. "Title of Work." Name of Site. Date of Posting/Revision, Organization. Date of Access, <URL>.

For example:

Cann, Robert. "Teaching Reading." Reading Journal. 7 Oct 2002. The Reading Association. 10 Nov 2002, <http://www.readingjournal.com/2002/ed12/article8.htm>.

Copying an Image from a Website

To copy an image, right click on it and select Copy. In the copy dialog box, select either **Picture or Picture Plus**, then click **OK.** The image has been placed on your computer clipboard. Open an application, such as Power Point; click on the Edit menu and then click **Paste.**

Saving a Sound File from a Website:

After obtaining copyright permission to save a sound file from the World Wide Web, right click on the sound link file to display a shortcut menu and then click **Save Target As**. A dialog box appears. (If you want to rename the sound file, do not delete or change the three letters that follow the period, i.e., the extension.) Then click **Save.** (Be sure to include the source information in your bibliography.) Microsoft PowerPoint 2000 supports sound files with the following extensions: wav, mid, rmi, aif, aifc, aiff, au.

Saving a video clip from a Website

Saving a video clip is similar to saving a sound file. Right click on the video link file to display a shortcut menu and then click **Save Target As.** (If you want to rename the video file, do not delete the extension.) Then click **Save.** (Again, be sure to include the source information in your bibliography.) Microsoft PowerPoint 2000 supports: avi, mov, mpg, mpeg, midi, mp2, cda, m1v, m3u, aif, au, m3d, cmr, prp, lit, flc, fli, flx.

Appendix C:
LOGO Teaching Methods

Lesson 1: Using an object with which to think: programming in the immediate mode:

New Commands for this lesson:

FORWARD #	OR	FD #
BACKWARD #	OR	BK #
RIGHT #	OR	RT #
LEFT #	OR	LT #

When a third grade student at the Moscow Charter School begins to program with LOGO, they have already had experience with programming through the use of the *Roamer Robot* software and the simulation software, *Crystal Rainforest* and *Mission Control.* At the beginning, we spend a short period of time teaching LOGO programming in the immediate mode. In this mode, a student types commands into the listener or bottom area of the programming screen, and the robot turtle immediately executes each command as it is typed. The process begins with the students learning to program a square in the immediate mode, using the following protocol:

FD 100 (the number 100 is variable)
RT 90 (the number is a constant)
FD 100
RT 90
FD 100
RT 90
FD 100
RT 90

Lesson 2: Writing a simple procedure (program using basic geometric shapes and symmetry:

New Commands for this Lesson:

REPEAT

Note: Procedure names are chosen by the programmer. The name can have no more that 8 characters and should have meaning.

A. Define a polygon: A polygon is formally defined as an enclosed object made of three or more straight lines.

B. Write a basic procedure to draw a polygon and then debug the procedure. A basic procedure can be named anything that has eight characters or less. The word "TO" must be in front of the procedure name. To rewrite our square program in the programming mode instead of immediate mode, see the commands listed below:

```
TO SQ
    FD 80
    RT 90
    FD 80
    RT 90
    FD 80
    RT 90
    FD 80
    RT 90
END
```

C. Use the repeat command. The repeat command allows the user to program a symmetrical polygon without repeating so many commands. To rewrite the square program with the repeat command, see the code below:


```
TO SQ
    REPEAT 4 [FD 100 RT 90]
END
```

Students are asked to count the sides on an object to determine the number of repetitions needed in the procedure. Looking at a rectangle reinforces the concept because teachers explain that on a rectangle the short sides and the long sides are repeated twice.

```
TO REC
    REPEAT 2 [FD 30 RT 90 FD 70 RT 90]
END
```

The repeat command is reviewed in relation to drawing multi-sided polygons such as triangles, pentagons, and hexagons. Code for other basic polygons that illustrate the repeat command include:

```
TO TRI
  REPEAT 3 [FD 100 RT 120]
END

TO PENTAGON
    REPEAT 5 [FD 50 RT 72]
END
```

D. Determine an angle with a total of 360 degrees on all sides. Multiplication and division can be used to determine angles for multi-sided figures. The turtle must turn a total of 360 degrees from the beginning to the end of any closed figure, thus the number used with the repeat command multiplied by the angle number must equal 360 degrees. Thus, to determine any angle for an equilateral polygon, divide 360 by the number of sides of the figure.

Lesson 3: Approaching problem-solving in a systematic way: Good programming strategies and debugging:

New Commands for this Lesson:

PENUP or PU

PENDOWN or PD

A. Break down the problem into small components. Students are taught to write a procedure in a systematic way to facilitate the debugging process. We encourage them to break their problem down into small components and write a separate program for each component. Also, writing a few lines of code and then testing them helps identify "bugs" or problems early on. Teaching students to use this process eliminates the problem of their having to identify "bugs" in a large amount of code.

B. Think and organize strategically. Students are taught how to break down the project into a series of smaller units. They are taught to list the smaller procedures that will be contained within a larger one. For example, to code a HOUSE procedure, the student can use a square for the house, a triangle for the roof, and rectangle for the door. The HOUSE procedure may include the following sub-procedures:

```
SQ
TRI
REC
```

C. Nesting procedures. LOGO users can call a procedure from within another procedure to create a larger program. For example, to write a program for a house, the sub-procedures that have already been written can be used as follows (this procedure has bugs that should be worked out by the student):

```
TO HOUSE
    SQ
    TRI
```



```
    REC
END
```

The following modified HOUSE procedure with no bugs enables users to insert a previously-written procedure using the PENUP (PU) and PENDOWN (PD) command. PENUP picks the turtles pen up so that it can change locations on the screen with drawing a line. PENDOWN puts the pen back down so that the turtle can draw as it carries out the procedure.

```
TO HOUSE
SQ
FD 80 RT 30
TRI
RT 60 FD 80 RT 90 FD 80 RT 90 FD 35
REC
END
```

Lesson 4: Screen Coordinates. This lesson introduces the concept of using coordinates to identify a specific point on a map by using an X and Y axis. Students are taught to move the turtle to a specific location on the screen by using the coordinate commands listed below:

New Commands for this lesson:

SETXY [# #]

SETX #

SETY #

SETH # (set heading)

Students can move the turtle to any position on the screen with fewer commands by defining specific coordinates using the SETXY command. The set heading command determines the direction the turtle is facing.

```
TO HOUSE
```

```
    SQ
    SETXY [0 80] SETH 30
    TRI
    SETXY [35 0] SETH 0
    REC
END
```

This lesson also introduces the concept that there are different means to accomplish the same end. For example, a comparison can be made between programming a procedure that uses SETXY and one that does not. The commands PENUP and PENDOWN can also be introduced to teach this concept.

Lesson 5: Using Variables. Variables can be used for any element of a simple procedure. For example, students are taught to define variables that vary the size of the square they've programmed:

```
TO SQ :SD
    REPEAT 4 [FD :SD RT 90]
END
```

To run a procedure with a variable, students assign a number to the variable, for example, SQ 25. The number 25 will be associated with the SD variable. Thus, the program will read by the computer as follows:

```
TO SQ
    REPEAT 4 [ FD 25 RT 90]
END
```

The following is a procedure that will code any equilateral polygon except a rectangle. The variables are :R for repeat, :D for distance, and :A for angle:

```
TO ANYSHP :R :D :A
    REPEAT :R [FD :D RT :A]
```



```
END
```

To change this procedure to draw a square, use this code and assign the following numbers:

```
ANYSHP 4  100  90
```

Lesson 6: Recursion or Looping: Using "IF THEN" logic. To repeat a shape, users can use recursion.

New Command for this lesson:

IF THEN

To program a shape that grows with each repetition, students are taught the following code:

```
TO GROWSQ :SD
    SQ :SD
    GROWSQ :SD + 5
END
```

To stop the recursion process, IF THEN logic is introduced with the following code:

```
TO GROWSQ :SD
    SQ :SD
    IF :SD > 250 THEN STOP
    GROWSQ :SD + 5
END
```

Lesson 7: Interactive Programming

New commands in this lesson:

```
PRINT [          ] OR PR [          ]
MAKE "VARIABLE
READLIST
```

By using PRINT, MAKE AND READLIST commands students learn to write code that will ask users questions to which they may respond. Code for a sample HELLO program with a variable called NM would include:

```
TO HELLO
    PR [WHAT IS YOUR NAME?]
    MAKE "NM READLIST
    PR [HELLO]
    PR :NM
    PR [HOW ARE YOU?]
END
```

The PRINT command will print whatever is in brackets on the screen. The MAKE READLIST commands, read the input from the user and assign it to a variable. An IF THEN statement can be added to produce an interactive program that responds to a user's input.

```
TO HELLO
    PR [WHAT IS YOUR NAME?]
    MAKE "NM READLIST
    PR [HELLO]
    PR :NM
    PR [HOW ARE YOU?]
    F :NM = [GOOD] THEN PRINT [LETS BEGIN] (CALL
    THE NEXT PROCEDURE HERE)
    IF :NM = [BAD] THEN PRINT [GOODBYE] STOP
END
```

Lesson 8: Arrays. This command outputs a list of numbers that describing the dimensions of the array.

New commands for this lesson:

ARRAYDIMS

A sample procedure is:

```
TO DIMARRAY
MAKE " ARRAY [ 3 4 5]
```

Lesson 9: Designing an Original Program. At this point, students understand enough commands to design their own programs. Sample programs that Moscow Charter School students use include the following:

- A brief tutorial about a specific topic, such as basketball, which includes four test questions
- A guessing game that gives clues to the user and encourages them to guess the correct answer
- A maze that allows the user to navigate with the following keys:

  F for forward

  B for backward

  R for right

  L for left

With the conclusion of this lesson, students have a basic grasp of the fundamental procedures they need to create meaningful and sophisticated procedures, and they are asked to design their own interactive procedure that someone else can test. Before writing the procedures, however, students are expected to draw a storyboard depicting each screen of the program, as it will appear to the user. Storyboards should also include answer-processing descriptions. See Appendix D for a sample of our storyboard handout.

Lesson 10: Programming an Original Design. To begin, students are asked to break their program down into logical sub-procedures, which are written and then debugged before they are added to the main procedure. Students are asked to list their sub-procedures on a handout that is attached to their storyboard. The list of sub-procedures and the storyboard guide students through their programming project. See Appendix D for a sample story board handout and Appendix E for a sample of the sub-procedures handout.

The final "main program" will call (list) these sub procedures in a logical order. Sub-procedures should be added to the main procedure one at a time and immediately debugged before adding another. Students are asked to fill out the handout in Appendix E to attach to their storyboard. These combined handouts serve as their working document to guide them through their programming project.

Appendix D:
Sample of Storyboard Handout

Student sketch and text that illustrate the contents of each screen.

1.

Student sketch and text that illustrate the contents of each screen.

2.

Student sketch and text that illustrate the contents of each screen.

3.

Appendix E:

Sample of Sub-Procedures Handout

Name:

Main program description (what will your program do?):

What are the different components of your program?

Please identify and list all of the sub-procedures that your will need to complete your programming project:

1.

2.

3.

4.

5.

6.

7.

8.

9.

10.

Notes:

Notes: